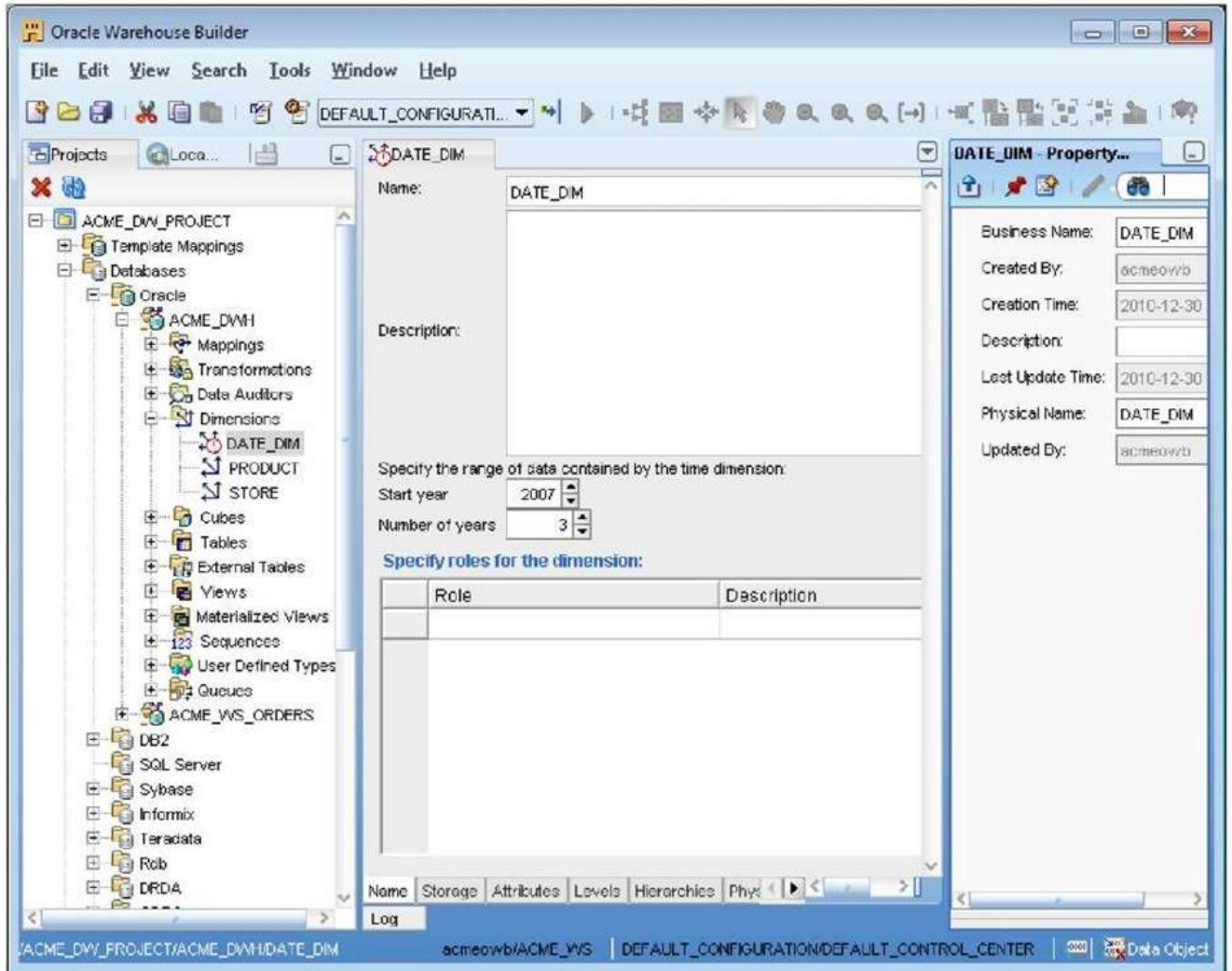


We've mentioned the Table Editor previously which is the data object editor for editing tables. We used it in topic 2 to create our source metadata definitions for the acme_pos transactional database, so let's take this opportunity to look a little closer at the editors for a dimension and a cube. We'll also discuss the overall Design Center interface and some of the other windows available to us in that interface as we're editing objects.

The object editors are the manual editor interfaces that the Warehouse Builder provides for us to create and edit objects. We did not have to use one to create a dimension, but more advanced implementations would definitely need to make use of it; for instance, to edit the cube to change the aggregation method that we just discussed. We'll take a brief look at editors here before moving on to get an idea of some of the features it provides.

We can get to a data object editor from the Project Navigator by double-clicking on an object, or by highlighting an object (by selecting it with a single click), and then right clicking and selecting Open from the menu. Editors in this latest release are now integrated into the main Design Center interface instead of popping open in a separate window. When editing any object now, a window appears in the Design Center containing the details to edit for the object. Let's open the date_dim dimension and examine the overall interface as shown here:



Your screen may look differently depending on what windows are open. The previous image depicts the Navigator window on the left which is displaying the Project Navigator, the main Editor window in the middle displaying the DATE_DIM dimension we just opened and the Property Inspector window on the right displaying properties for the DATE_DIM dimension. Any of these windows can be opened, closed, minimized, or relocated offering tremendous flexibility in laying out our working area. If a window is taking up space and we don't need it at the moment, just minimize it by clicking the minimize icon in the upper right corner of the window if that option is available. We can also close any window we want by hovering the mouse over the window title and clicking the X that appears or by right clicking over the window title and selecting Close from the popup. The main Editor window cannot be minimized but can be closed.

A complete discussion of all the windows available is in the Warehouse Builder Concepts Guide available at [http:// download.oracle.com/docs/cd/E118 8 2_01/owb.112/ e10581/toc.htm](http://download.oracle.com/docs/cd/E11882_01/owb.112/e10581/toc.htm)

There is a complete topic devoted to a tour of the user interface that will provide more details than we can cover here. Use the following URL to jump right to that: http://download.oracle.com/docs/cd/E11882_01/owb.112/e10581/uitour.htm#BABFDIHH

We'll briefly discuss the windows we can see in the image above as well as some of the additional available windows.

- **Navigator Windows:** We discussed the windows available for navigation in the last topic, the Projects, Locations, and Global Explorer windows that show on the left in the above image. Each of these windows can be displayed or hidden as needed and if not visible, can be displayed by selecting View from the main menu and then choosing the window to display.

For any of these windows we'll discuss, if they are not currently visible, they can be displayed by selecting them from the View main menu so when we discuss any windows below that are not visible, that is how we can display them.

- **Canvas:** This is not really a separate window but is a feature of some editors such as the Mapping Editor, that provides an area in which the contents are displayed graphically. Each object is displayed in a box with the name of the object as the title of the box and attributes of the object listed inside the box. These boxes can be moved around and resized manually to suit our tastes. We'll see an example next when we look at an editor window.

- **Property Inspector:** This window is visible in the above image and is available for setting various properties of objects that have been selected in the Navigator windows or in the editors. It is context sensitive and will display or not display any properties depending on whether there are properties pertaining to the selected object.

- **Configuration:** The configuration window displays configuration information about a selected item. Display the Configuration window by right clicking an object in the Project Navigator and selecting Configure... from the pop-up menu. It is here that we can change the deployment option for the object to deploy OLAP metadata if we want a relational implementation to store the OLAP metadata. With the date_dim dimension selected in the Project Navigator, right click on it and select Configure and in the Configuration window click the plus sign beside the Identification section to expand it. It contains a setting for the deployment option and we can see that it is set to deploy data objects only. For dimensions, the options are to deploy to catalog only (the OLAP catalog), deploy data objects only, or deploy all to do both. For cubes, there is an additional option to deploy aggregations.

- **Component Palette:** The Palette is primarily applicable to editing mappings and provides a list of the items that can be dragged and dropped to build a mapping. It contains each of the objects that can be used in the mapping. We can use this to create objects on our canvas in the editor by clicking and dragging to the canvas. This will create a new object where clicking and dragging from the Projects Navigator will place an already created object on the canvas. We'll discuss this more in detail when we start looking at mappings in the next topic.

- **Bird's Eye View:** This window displays a miniature version of the entire canvas and allows us to scroll around the canvas without using the scroll bars. It is applicable only for editors that display a graphical canvas of objects. We can click and drag the blue-colored box around this window to view various portions of the main canvas, which will scroll as we move the blue box. We will find that in most editors, we will quickly outgrow the available space to display everything at once and will have to scroll around to see everything. This can come in very handy for rapidly scrolling the window.

- **Editor:** This is the actual editor window that contains details about the object we are currently editing. It is the center window in the above image. Various tabs will appear, which display information for us depending on the object being edited. The tabs from left to right are as follows when editing a dimension some of which we can see in the above image:

- **Name:** This tab displays the name of the dimension along with some other information specific to the dimension type we are looking at. In this case, it's a Time dimension created by the Time Dimension Wizard and so it displays the range of data in our Time dimension.

- **Storage:** Here we can see what storage option is set for our dimension object in the database, whether Relational or Multidimensional. If we wanted to switch between the two, this is where we could do it. For a relational implementation, we're able to specify a star or snowflake schema and whether we want to create composite unique keys. A composite key is one made up of more than one column to define uniqueness for a record. In most cases, it is a good idea to have this checked as it enforces uniqueness in the database for our dimension records. It will not make a difference in our particular case for the test data we'll be using. For a dimension, it will use the business identifiers we've specified as the key fields.

- **Attributes:** The attributes tab is where we can see the attributes that are designed for our dimension. It displays the attributes in a tabular form allowing us to view and/or edit them, including adding new attributes or deleting the existing ones. It is here that we can also change the description of our attributes if we wanted, or add descriptions the wizard did not add.

You may have noticed by now that the attributes in our Time dimension are not editable. They all appear as one solid background. We can scroll the window to display them and see what they are set to, but we can't change them. This is a feature of the Time dimension that was created by the wizard. It has created extra objects (as we saw earlier) to support the Time dimension, such as a mapping that could break if the wrong changes are made. So, it disallows changes. It is possible to modify the dimension behind the scenes to edit things, but that is a much more advanced topic. As mentioned earlier, we could have also defined our time dimension using the regular dimension wizard and these would all be editable, but we wouldn't have the mapping created automatically to populate it.

- **Levels:** This is where we view and/or edit the levels for our dimension. We are able to edit some of the information on this tab for the Time dimension created by the wizard, but not all. We can check and uncheck boxes to indicate which of the various level types we want to use and which attributes are applicable to which level, but that is it. We are not able to add or remove any levels or attributes.

If we were to view one of the other dimensions we created, it would be fully editable. For those other dimensions we could also assign different names and descriptions to the attributes for each level.

° **Hierarchies:** This tab will let us specify hierarchy information for our dimension and will even let us create a new hierarchy. It's possible that we may have selected more levels on the previous page and now need to assign them to a hierarchy. There is also a Create Map button here that will automatically generate the mapping for us if we modify the hierarchies. This is one of the benefits of the Time dimension created by the wizard. Ordinary dimensions such as our Store and Product dimension will not have this Create Map button displayed on their Hierarchies tab.

° **SCD:** This tab is for specifying the Slowly Changing Dimension policy to use. The Time Dimension will not show this tab because its not applicable to that type of dimension since the contents are preloaded and won't change.

° **Orphan:** This tab will also not be available for the Time dimension but will be for the Product and Store dimensions. This is a new feature in the 11gR2 release of the Warehouse Builder that provides an automated way to manage what to do about dimensional records (dimensions and cubes) that can't be loaded because one or more records do not have a parent record. This is also known as Early Arriving Facts in industry and Orphan Management is OWB's answer to addressing this issue. Since we're working with a known set of canned test data, there aren't any orphans to worry about but for actual implementations this is a feature that can greatly improve productivity in loading data. For dimensions we have three main categories of Orphan Management to specify, for loading the dimension, removal of records from the dimension, and whether to deploy an error table for orphans.

There are three main options available for handling orphans:

- **No Maintenance:** This is the default setting and indicates that orphans will not be detected and nothing will be done about them.
- **Default Parent:** This setting indicates that a default parent record should be used if one is null or invalid and if a default parent doesn't exist, it will create one to use. There is a Settings button available to set the default level row to use.
- **Reject Orphan:** This will cause the orphan record to be rejected.

For more information, on the new Orphan Management policies available in the Warehouse Builder, see topic 3 in the Warehouse Builder Data Modeling, ETL and Data Quality Guide at the following URL: [http:// download.oracle.com/docs/cd/E118 8 2_01/owb.112/e10935/](http://download.oracle.com/docs/cd/E11882_01/owb.112/e10935/)

[dim_objects.htm#BABEJGDC](http://download.oracle.com/docs/cd/E11882_01/owb.112/e10935/dim_objects.htm#BABEJGDC)

There is a blog entry in the Oracle Warehouse Builder Blog that talks specifically about Early Arriving Facts and Orphan Management which can be accessed for more details at the following URL:

http://blogs.oracle.com/warehousebuilder/2010/06/owb_11gr2_-_early_arriving_facts.html

◦ **Physical Bindings:** This tab displays a canvas with the dimension represented graphically along with the underlying physical table showing how attributes of the dimension are mapped (or bound) to the table columns.

• **Data Viewer:** The Data Viewer is a more advanced feature that allows us to actually view the data in an object we are editing. This is only available for an object if it has been deployed to the database and has data loaded into it. It has a query capability to retrieve data and can specify a where clause to get just the data we might need to see. For relational implementations, it will not display the data for a dimension or cube; but we can use it to view the data in the underlying table. It is accessible from the View menu by selecting the Data... menu entry when editing a data object.

• **Cube Editor:** If we edit the Sales Cube, the editor window has a slightly different set of tabs available to it which we'll cover briefly here:

◦ **Name:** It has a name tab like the dimensions to display its name.

◦ **Storage:** It has a storage tab as per dimensions. However, we see a different option here under the Relational (ROLAP) option where we can create bitmap indexes. An index is a database feature that allows faster access to data. It is somewhat analogous to the index of a topic that allows us to get to a page in the topic with the information we want much faster. A bitmap-type index refers to how it is stored in the database and is generally a better option to use for data warehouse implementations (so it is checked by default). There is also a composite unique key checkbox for cubes as there was for dimensions.

For a cube, checking this box will create a unique key out of the foreign keys for the dimensions referenced by the cube. We want to check this box to ensure we can't enter duplicate data into our cube, that is, more than one cube record with the same set of dimension attributes assigned.

◦ **Dimensions:** Instead of attributes, the cube has a tab for dimensions. The dimensions referenced by a cube are basically its attributes.

◦ **Measures:** The next tab is for the measures of the cube. It is for those values that we are storing in our cube as the facts that we wish to track.

◦ **Aggregations:** Instead of hierarchies, a cube has aggregations. There are various methods of aggregation that we can select, as seen in the drop-down box, the most common of which is sum, which is the default. This is where the default aggregation method referred to earlier can be changed. There will be no aggregations in a pure relational implementation, so we will leave this tab set to the defaults and not bother changing it.

◦ **Orphan:** This feature is available for cubes also and provides an automated way to manage what to do about cube records that can't be loaded because one or more dimension records cannot be found for it. There are two options to account for here, what to do if a dimension key is null and what to do if a dimension key is found to be invalid. In each case there are three options to choose from for how to handle the orphans:

1. No Maintenance: This is the default setting and indicates that orphans will not be detected and nothing will be done about them.

2. Default Dimension Record: This setting indicates that a default dimension record should be used if one is null or invalid and if a default dimension doesn't exist, it will create one to use. Default dimension records can be created from the Orphan tab of the dimension as discussed above.

3. Reject Orphan: This will cause the orphan record to be rejected.

These are the main features of the Design Center interface and editors for dimensions and cubes. We can use it to view the objects the wizards have created for us, edit them, or create brand new objects from scratch. We can start with an empty canvas and drag new objects from the palette, or existing objects from the explorer, and then connect them. We will see other editors very similar to this from the next topic when we start to look at ETL and mappings.

Summary

In this topic, we dove right into creating our three dimensions and a cube using the Warehouse Builder Design Center. We used the wizards available to help us out, as well as investigated the flexibility to manually create, view, and edit objects using the data object editors for dimensions and cubes. In a relatively short amount of time, we were able to design a data warehouse structure that could be used as is, or expanded to support more detailed information.

Now that we have our sources defined and our targets designed, it's time to start thinking about loading that target. Next, we'll look at some Extract, Transform, and Load (ETL) basics to lay the groundwork for designing the ETL we'll use to actually load data into our data warehouse.